

Fondamenti di Informatica C

Esercitazioni di Laboratorio / 1

<http://polaris.ing.unimo.it/fiC/laboratorio.html>

Ing. Francesco De Mola

demola.francesco@unimore.it

→ Ricevimento su appuntamento ←

DII, Modena – Via Vignolese
(lab. Dottorandi 1° piano)
Tel.: 059 205.61.42

DISMI, Reggio Emilia – Via
Amendola 2 (Pad. Morselli)
Tel.: 0522 52.26.60

Home Page:

<http://www.agentgroup.ing.unimo.it/didattica/curriculum/francesco>

Outline

- Astrazione di dato e Tipo di dato astratto (ADT) - *ripasso*
- Implementazione di un dato astratto e di un tipo di dato astratto per la gestione di un **conto corrente** in C
- Esempio di **implementazione OOP** mediante il linguaggio C non orientato agli oggetti

Astrazione di Dato vs. ADT

- **ASTRAZIONE DI DATO:**

dato che definisce una entità astratta, rappresentante un concetto della realtà, con

- dati (o “attributi”)
- e operazioni (o “comportamenti” o “metodi”) che si possono effettuare sui dati

- **TIPO DI DATO ASTRATTO (ADT):**

tipo da cui è possibile definire (“**istanziare**”) variabili come astrazioni di dato.

Fondamenti di Informatica C - Esercitazione 1

3

Counter con Astrazione di Dato

```
→ count.h:
int getValue(void);
void setValue(int);
void inc(void);
void dec(void);

→ count.c:
static int Contatore = 0;
int getValue(void)
{return Contatore;}
void setValue(int v)
{Contatore=v;}
void inc(void)
{Contatore++;}
void dec(void)
{Contatore--;}

→ Uso:
#include "count.h"
void main(){
    setValue(10);
    printf("Valore = %d",
        getValue());
    inc();
}
```

Counter con ADT

```
→ counter.h:
typedef Contatore int;
int getValue(Contatore);
void setValue(Contatore*, int);
void inc(Contatore *c);
void dec(Contatore *c);

→ counter.c
#include "counter.h"
int getValue(Contatore c)
{return c;}
void setValue(Contatore* c, int v)
{*c=v;}
void inc(Contatore *c){ (*c)++;}
void dec(Contatore *c){ (*c)--;}

→ Uso:
#include "counter.h"
void main(){
    Contatore c1, c2;
    setValue(&c1, 10);
    setValue(&c2, -31);
    printf("Valore di c1 = %d",
        getValue(c1));
    inc(&c2); dec(&c1);
}
```

4

Esercizio: Conto corrente bancario come astrazione di dato e come tipo di dato astratto

Il conto corrente prevede i seguenti dati:

- **numero di CC** (intero)
- **saldo attuale** (intero)
- **lista dei movimenti** effettuati sul conto (vettore di interi)
- **contatore dei movimenti** effettuati (intero)

e le seguenti operazioni (eseguibili sui dati):

- **inizializzazione del conto**: definizione del numero del CC, inizializzazione del saldo
- **prelievo**: prelievo di un certo importo e registrazione dello stesso
- **versamento**: versamento di un certo importo e registrazione dello stesso
- **restituzione** del saldo
- **stampa della lista dei movimenti**: stampa a video di tutti i movimenti effettuati

Nel vettore dei movimenti vengono registrati gli importi di prelievo e versamento (1000,-500,800,...). Il contatore dei movimenti conta il numero dei movimenti registrati nel vettore. Scrivere un main di prova per mostrare l'uso del dato con le funzionalità implementate.

Fondamenti di Informatica C - Esercitazione 1

5

Soluzione: Conto corrente bancario come astrazione di dato e come tipo di dato astratto

- <http://polaris.ing.unimo.it/fiC/laboratorio.html>

- Astrazione di dato: `cc.h`
`cc.c`
`main1.c`

- Tipo di dato Astratto: `ccADT.h`
`ccADT.c`
`main2.c`

Fondamenti di Informatica C - Esercitazione 1

6

Vantaggi vs. Svantaggi

Astrazione di Dato

- **Pro:**
il dato è protetto,
essendo accessibile
dall'esterno solo
mediante le funzioni
definite
- **Contro:**
non è possibile
definire più istanze del
dato

Fondamenti di Informatica C - Esercitazione 1

ADT

- **Pro:**
possibilità di definire
più istanze del dato
- **Contro:**
il dato è accessibile
dall'esterno, quindi
si possono eseguire
su di esso
operazioni non
ammesse

7

Implementazione conto corrente bancario tramite Object Oriented Programming in C

Si cerca di ottenere un'implementazione **Object Oriented** usando un linguaggio **NON Object Oriented**, attraverso l'uso di strutture pubbliche e private.

- <http://polaris.ing.unimo.it/fiC/laboratorio.html>
- **CC_pub.h**
CC_private.c
Main1.c
- Variante con strutture annidate: **CC_private2.c**

Fondamenti di Informatica C - Esercitazione 1

8